CSC 311

Midterm Questions & Answers

**Q1:[40 points= 10+10+10+10]**

(a) Using the definition of $\Omega$, prove the following by finding $c$ and $n_0$:
$3n^4 - 4n^2 \in \Omega(n^4)$.

(b) Using the definition of $O$, prove the following by finding $c$ and $n_0$:
$\log(n!) \in O(n \log n)$.

(c) For two constants $a, b \geq 1, a \neq b$, do $a^n$ and $b^n$ have the same order of growth? why?

(d) Find a $\Theta$ estimate for the function $M(n)$, which is defined as follows:
$M(1) = 1, M(n) = M(\lfloor \frac{n}{2} \rfloor) + n^2$.

**Q2:[30 points=6+12+12]**

Consider the following algorithm, where $A$ is an array of $n$ integers and indexed from 1 to $n$. $L$ and $R$ are indexes of the array $A$. The first call to the algorithm will be $S(A[1..n])$.

```
Algorithm S(A[L..R])
if (L > R) then
    return 0;
else
    M:= 1;
    for i = 1 to R do
        M:=M * A[R];
    end
    return S(A[L..R − 1]) + M;
end
```

(a) What does this algorithm do? Give an example.

(b) Set up a recurrence relation for the basic operation count as a function of the input size $n$.

(c) What is the time complexity of this algorithm? Justify your answer.

**Q3:[30 points=10+20]**
A circular shift operation on an array moves each item to the next location and the last item is moved to the first location. For example a circular shift to the array $[1, 5, 9]$ would result in the array $[9, 1, 5]$. You are given an array of $n$ distinct integers and you are told that the array was initially sorted in an increasing order and then $k$ circular shift operations were applied to the array $(0 < k < n)$.

(a) Give the pseudocode of a brute force algorithm to find $k$.

(b) Give the pseudocode of an algorithm that finds $k$ in $O(\log n)$ time.

**The Master Theorem:**
Let $a \geq 1$ and $b > 1$ be constants, let $f(n)$ be an asymptotically positive function, and let $M(n)$ be defined on the nonnegative integers by the recurrence:
$M(n) = aM(\frac{n}{b}) + f(n)$,
where we interpret $\frac{n}{b}$ to mean either $\lfloor \frac{n}{b} \rfloor$ or $\lceil \frac{n}{b} \rceil$. Then $M(n)$ can be bounded asymptotically as follows.

1. If $f(n) \in O(n^{\log_b a - \epsilon})$ for some constant $\epsilon > 0$, then $M(n) \in \Theta(n^{\log_b a})$.
2. If $f(n) \in \Theta(n^{\log_b a} \log^k n)$, with $k \geq 0$, then $M(n) \in \Theta(n^{\log_b a} \log^{k+1} n)$.
3. If $f(n) \in \Omega(n^{\log_b a + \epsilon})$ for some constant $\epsilon > 0$ AND $af(\frac{n}{b}) \leq cf(n)$ for some constant $c < 1$ and all sufficiently large $n$, then $M(n) \in \Theta(f(n))$.

Q1

a) $3n^4 - 4n^2$

$\geq 2n^4$ , whenever $n^4 \geq 4n^2$

$n^2 \geq 4$

$n \geq 2$

So, $c = 2$ , $n_0 = 2$

---

b) $\log(n!) = \log(n*(n-1)*(n-2)* \cdots *2*1)$

$= \log(n) + \log(n-1) + \log(n-2) + \cdots \log(2) + \log(1)$

$\leq \log(n) + \log(n) + \log(n) + \cdots \log(n) + \log(n)$

$= n \log n$

$c = 1$ , $n_0 = 2$

---

c) $\lim_{n \to \infty} \frac{a^n}{b^n} = \lim \left(\frac{a}{b}\right)^n = \begin{cases} \infty & \text{if } a > b \\ 0 & \text{if } a < b \end{cases}$

so their order of growth is not the same.

---

$\boxed{Q_1}$

d) <u>Option 1</u>: Using Backward Substitutions

$$M(n) = M\left(\frac{n}{2}\right) + n^2 \qquad , \boxed{n = 2^k}$$

$$M(2^k) = M(2^{k-1}) + 2^{2k}$$

$$= M(2^{k-2}) + 2^{2(k-1)} + 2^{2(k)}$$

$$= M(2^{k-3}) + 2^{2(k-2)} + 2^{2(k-1)} + 2^{2(k)}$$

$$\vdots$$

$$= M(2^{k-k}) + 2^{2(1)} + 2^{2(2)} + \dots 2^{2k}$$

$$\cancel{= M(1) + 4 + 4^2 + 4^3 + \dots + 4^k}$$

$$= M(1) + 2^2 + 2^4 + 2^6 + \dots 2^{2k}$$

$$\leq 1 + \left(2^{2k+1} - 1\right) \qquad \boxed{\sum_{i=1}^{k} 2^i = 2^{k+1} - 1}$$

$$= 2^{2k+1}$$

$$= 2 \cdot (2^k)^2 = 2 \cdot n^2$$

$$\in \Theta(n^2)$$

Q1

d) Option 2 : Using the Master Theorem

$a = 1$, $b = 2$ $\log_b a = 0$, $f(n) = n^2$

Case 3 applies b/c:

① $n^2 \in \Omega(n^{0+\epsilon})$ ✓ $\epsilon \leq 2$

② $a f(\frac{n}{b}) = a \frac{n^2}{b^2} \leq c n^2$, $c = \frac{1}{2}$ ✓

So $M(n) \in \Theta(f(n))$

$\in \Theta(n^2)$

Q2

a) $\sum_{i=1}^{n} (A[i])^i$

Ex:

| 1 | 2 | 3 |
|---|---|---|
| 4 | 2 | 3 |

$$\text{Output} = 4^1 + 2^2 + 3^3$$
$$= 4 + 4 + 27 = 35$$

---

b)

$$M(n) = \begin{cases} 1 & , n=1 \\ M(n-1)+n & , n>1 \end{cases}$$

---

c) $M(n) = M(n-1) + n$

$$= M(n-2) + (n-1) + n$$
$$= M(n-3) + (n-2) + (n-1) + n$$
$$\vdots$$
$$= M(1) + 2 + 3 + 4 + \cdots + (n-1) + n$$
$$= 1 + 2 + 3 + \cdots + (n-1) + n$$
$$= \sum_{i=1}^{n} i = \frac{n(n+1)}{2} \in \Theta(n^2)$$

Q3

We should return the index of the minimum integer in the array. And the array was initially sorted.

a)

```
Shift (A[0 .. n-1])
{
    i = 0;
    While (A[i] < A[i+1])
        i = i+1;

    return i+1;
}
```

Ex:

| 0 | 1 | 2 |
|---|---|---|
| 9 | 1 | 5 |

i = 0   exit the loop and return 1

Ex:

| 0 | 1 | 2 |
|---|---|---|
| 5 | 9 | 1 |

i = 0
i = 1   exit the loop & return 2

Q3

b)

```
Shift (A[0 .. n-1], L, R)
{
   if (L == R)
        return L
   else
        m = ⌊(L+R)/2⌋

        if (A[m] < A[R])
           return Shift(A[], L, m);
        else
           return Shift(A[], m+1, R)
}
```